

## **AMENDMENTS**

In accordance with 37 CFR §1.121, please amend the above-identified application as set forth below.

### ***Amendments to the Claims***

Please amend the claims as set forth below.

1. (Currently Amended) A method for peer-to-peer messaging between network resources comprising:

communicating with a first process by writing a first text file in a first scratch space, where the first text file describes at least one of ~~at least~~ a first set of information that a second process has generated and ~~at least~~ a first action to be performed on the first set of information;

detecting, by a first arbiter, the first text file, wherein the first arbiter is implemented by the first process; ~~and~~

performing at least one of:

implementing, by the first arbiter, the first action, and

applying, by the first arbiter, logic embedded within the first arbiter to determine

actions to be performed on the first text file;

communicating with a third process independently from a central master control system by writing a second text file in a second scratch space, wherein the second text file describes at least one of a second set of information that the first process has displayed and a second action to be performed on the second set of information;

detecting, by a second arbiter, the second text file, wherein the second arbiter is implemented by the third process; and

performing at least one of:

implementing, by the second arbiter, the second action, and

applying, by the second arbiter, logic embedded within the second arbiter to determine actions to be performed on the second text file.

2. (Previously Presented) A method in accordance with claim 1 wherein communicating with the first process comprises communicating with the first process by writing an American standard code for information exchange (ASCII) file, and wherein the first arbiter reviews the first text file and upon determining that the first text file includes a match with the first process, performs at least one of the implementing and applying steps.

3. (Original) A method in accordance with claim 2 wherein communicating with the first process by writing the ASCII file comprises communicating with the first process by writing one of a hypertext markup language (HTML) file, an extensible HTML (XML) file, a multipurpose internet mail extensions (MIME) file, a .NET file, and a simple object access protocol (SOAP) file in the first scratch space

4. (Original) A method in accordance with claim 1 wherein applying, by the first arbiter, logic embedded within the first arbiter comprises at least one of: moving the first text file to a second scratch space; moving the first set of information to the second scratch space; and obtaining index information from the first text file and moving images associated with the index information into a storage repository.

5. (Original) A method in accordance with claim 1 further comprising encoding the first set of information within the first text file.

6. (Original) A method in accordance with claim 1 further comprising referencing the first set of information as being in an external file.

7. (Original) A method in accordance with claim 1 further comprising: enabling, by the first process, an input from a user; and writing the input to at least one of the first text file and a second text file in the first scratch space.

8. (Original) A method in accordance with claim 1 wherein communicating with the first process comprises communicating with an image display process by writing the first text file in the first scratch space.

9. (Original) A method in accordance with claim 1 wherein communicating with the first process comprises communicating with the first process by writing the first text file in the first scratch space, wherein the first text file describes at least one of an image that a scanning process has generated and the first action to be performed on the image.

10. (Original) A method in accordance with claim 1 further comprising: reading, by the first arbiter, instructions within the first text file.

11. (Original) A method in accordance with claim 1 wherein applying, by the first arbiter, logic embedded within the first arbiter comprises determining whether data that is referenced by the first text file as being in a second text file should be processed.

12. (Currently Amended) A method in accordance with claim 1, wherein said processes are selected from the group consisting of an image scanning process, an image display process, an image editing process, a word editing process, a spreadsheet process, an e-mail process, a calendar process, a contact management process, a database process, a web browser process and any combination thereof, further comprising:

~~communicating with a third process independently from a central master control system by writing a second text file in a second scratch space, wherein the second text file describes at least one of a second set of information that the first process has displayed and a second action to be performed on the second set of information;~~

~~detecting, by a second arbiter, the second text file, wherein the second arbiter is implemented by the third process; and~~

~~performing at least one of:~~

~~implementing, by the second arbiter, the second action; and  
applying, by the second arbiter, logic embedded within the second arbiter to determine actions to be performed on the second text file.~~

13. (Original) A method in accordance with claim 1 further comprising: specifying a format of the first text file; and changing the format of the first text file to the specified format.

14. (Original) A method in accordance with claim 13 where changing the format of the first text file includes one of: converting the first text file from a plain text file to a hypertext markup language (HTML) file; and converting the first text file from a simple object access protocol (SOAP) to a NET file; and restructuring data within the first text file.

15. (Original) A method in accordance with claim 1 further comprising: requesting a public key from an authority; encrypting a portion of the first text file by using the public key; signing the portion; transmitting the portion and the public key to a second scratch space; and requesting an authentication of a second process that received the portion and the public key.

16. (Previously Presented) A method in accordance with claim 15 further comprising: reading the first text file received within the first scratch space to find a digital signature; requesting an authentication of the digital signature; further transmitting the portion from the second process to a service on obtaining the authentication of the second signature; decrypting the portion using a private key; and sending the decrypted portion from the service to the second process.

17. (Original) A method in accordance with claim 1 further comprising applying, by the first arbiter, at least one of a File Transfer Protocol (FTP), a Hypertext Transfer Protocol (HTTP), and a file services network protocol to move the first text file between network resources.

Claims 18-35 (Canceled)

36. (Currently Amended) ~~A method in accordance with claim 1~~ A method for peer-to-peer messaging between network resources comprising:  
communicating with a first process by writing a first text file in a first scratch space,  
where the first text file describes at least one of a first set of information that a second process  
has generated and a first action to be performed on the first set of information, wherein said first text file drives the first process on a first computer having a first operating system and wherein

said second process is on a second computer having a second operating system without messaging systems of said first operating system being resident on said second operating system;  
detecting, by a first arbiter, the first text file, wherein the first arbiter is implemented by the first process; and  
performing at least one of:  
implementing, by the first arbiter, the first action, and  
applying, by the first arbiter, logic embedded within the first arbiter to determine actions to be performed on the first text file.

37. (Currently Amended) A method for peer-to-peer messaging between network resources comprising:

communicating with a first process by writing a first text file in a first scratch space, where the first text file describes at least one of a first set of information that a second process has generated and a first action to be performed on the first set of information that the second process has generated;

detecting, by a first arbiter, the first text file, wherein the first arbiter is implemented by the first process independently from a master control system; and  
determining, by the first arbiter independently from a master control system, that the first text file includes a match with the first process, and performing at least one of:  
implementing, by the first arbiter independently from a master control system, the first action, and  
applying, by the first arbiter independently from a master control system, logic embedded within the first arbiter to determine actions to be performed on the first text file.

38. (Previously Presented) A method in accordance with claim 37 wherein said first text file drives the first process on a first computer having a first operating system and wherein said second process is on a second computer having a second operating system without messaging systems of said first operating system being resident on said second operating system.

39. (Currently Amended) A method in accordance with claim 37 wherein said processes are selected from the group consisting of an image scanning process, an image display process, an image editing process, a word editing process, a spreadsheet process, an e-mail process, a calendar process, a contact management process, a database process, a web browser process and any combination thereofarbitrarily performs said implementing and applying steps independently from a central master control system.

40. (Previously Presented) A method in accordance with claim 37 wherein applying, by the first arbiter, logic embedded within the first arbiter comprises at least one of: moving the first text file to a second scratch space; moving the first set of information to the second scratch space; and obtaining index information from the first text file and moving images associated with the index information into a storage repository.

41. (Currently Amended) A method in accordance with claim 37 further comprising determining, by the first arbiter independently from a central master control system, whether the first text file contains at least one of a token and an address matching a network resource with which the first arbiter is associated or resident, and moving the first text file between network resources by the first arbiter.

42. (Previously Presented) A method for peer-to-peer messaging between network resources comprising:

communicating with a first process by writing a first text file in a first scratch space, where the first text file describes at least one of a first set of information that a second process has generated and a first action to be performed on the first set of information that the second process has generated;

detecting, by a first arbiter, the first text file, wherein the first arbiter is implemented by the first process; and

performing at least one of:

implementing, by the first arbiter and independently from a central master control system, the first action; and

applying, by the first arbiter and independently from a central master control system, logic embedded within the first arbiter to determine actions to be performed on the first text file.

43. (Previously Presented) A method in accordance with claim 42 wherein the first arbiter reviews the first text file and upon determining that the first text file includes a match with the first process, performs at least one of the implementing and applying steps.

44. (Previously Presented) A method in accordance with claim 42 wherein said first text file drives the first process on a first computer having a first operating system and wherein said second process is on a second computer having a second operating system without messaging systems of said first operating system being resident on said second operating system.



45. (Previously Presented) A method in accordance with claim 42 wherein applying, by the first arbiter, logic embedded within the first arbiter comprises at least one of: moving the first text file to a second scratch space; moving the first set of information to the second scratch space; and obtaining index information from the first text file and moving images associated with the index information into a storage repository.

46. (Previously Presented) A method in accordance with claim 42 further comprising moving the first text file between network resources by the first arbiter.

47. (New) A method in accordance with claim 42, wherein said first arbiter performs at least one of said implementing and applying steps in an asynchronous manner relative to a second arbiter associated with the second process.

48. (New) A method in accordance with claim 1, wherein said first text file drives the first process on a first computer having a first operating system and wherein said second process is on a second computer having a second operating system without messaging systems of said first operating system being resident on said second operating system.

49. (New) A method in accordance with claim 36, wherein said detecting step is performed independently from a master control system and wherein said first arbiter performs at least one of said implementing and applying steps in an asynchronous manner relative to a second arbiter associated with the second process.